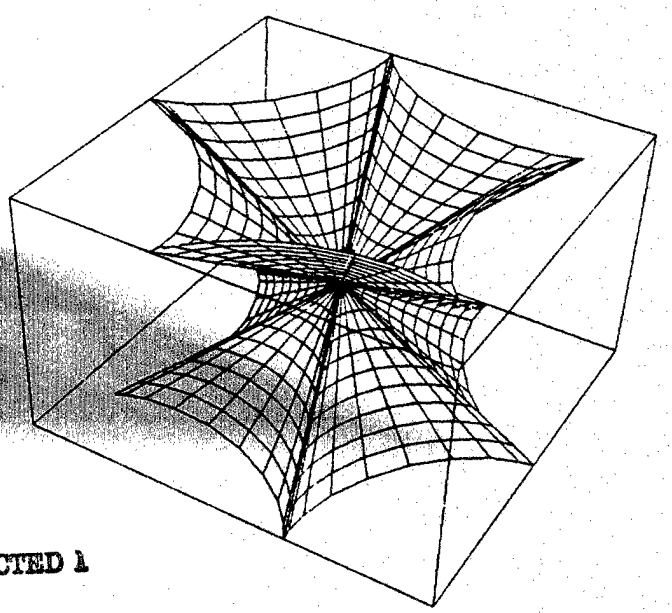
Center for Pure and Applied Mathematics

University of California at Berkeley

PAM– 671

WHAT HADAMARD MISSED

Beresford N. Parlett



March 1996

This report was done with support from the Center for Pure and Applied Mathematics. Any conclusions or opinions in this report represent solely those of the author(s) and not necessarily those of the Center for Pure and Applied Mathematics or the Department of Mathematics.

DEPARTMENT OF THE NAVY
OFFICE OF NAVAL RESEARCH
SEATTLE REGIONAL OFFICE
1107 NE 45TH STREET. SUITE 350
SEATTLE WA 98105-4631

IN REPLY REFER TO:

4330
ONR 247
11 Jul 97

From: Director, Office of Naval Research, Seattle Regional Office, 1107 NE 45th St., Suite 350, Seattle, WA 98105
To: Defense Technical Center, Attn: P. Mawby, 8725 John J. Kingman Rd., Suite 0944, Ft. Belvoir, VA 22060-6218

Subj: RETURNED GRANTEE/CONTRACTOR TECHNICAL REPORTS

1. This confirms our conversations of 27 Feb 97 and 11 Jul 97. Enclosed are a number of technical reports which were returned to our agency for lack of clear distribution availability statement. This confirms that all reports are unclassified and are "APPROVED FOR PUBLIC RELEASE" with no restrictions.

2. Please contact me if you require additional information. My e-mail is *silverr@onr.navy.mil* and my phone is (206) 625-3196.

ROBERT J. SILVERMAN

What Hadamard Missed

Beresford N. Parlett *

March 19, 1996

Abstract

Consider the task of finding all the eigenvalues of a dense matrix. We show how Hadamard's procedure (1891) can be organized into Aitken's H-table (1925) and how the H-table may be transformed into Rutishauser's qd-array (1953) with the help of the Lanczos algorithm. We show how the qd algorithm can be interpreted as defining the LR algorithm (1958). Finally we show how the original qd algorithm may be transformed into the shifted differential qd algorithm *dqds* developed by Fernando and Parlett (1993/94). The Lanczos algorithm takes a dense matrix into tridiagonal form and then *dqds* is a fast and accurate procedure for extracting the eigenvalues.

This paper evolved from talks given at UC, Berkeley, in October 1992, at the Symposium on Scientific Modelling, Charleston, Illinois and at the Mass. Inst. for Tech., both in October 1995.

*Mathematics Department and Computer Science Division, EECS Department, University of California, Berkeley, CA 94720, USA. Supported by ONR, Contract N000014-90-J-1372.

1 Introduction

The story to be told here is not strictly a contribution to the history of mathematics. It might be described, using a fashionable adjective, as virtual history. With the benefit of hindsight we trace the transformations of an idea from its original pre-computing form into a powerful and elegant algorithm that is ideally suited for implementation on a 'parallel' computing system that can keep many processors busy at the same time.

Our tale begins with the doctoral dissertation of the illustrious French mathematician Jacques Hadamard in 1891. His solution to the problem, described in the next section, yields a lousy algorithm. Our title intends no disparagement of that great man; neither he nor any of his contemporaries would have dreamt of *evaluating* the determinants he so cleverly introduced. Hadamard 'missed' the subsequent evolution of his idea because he saw no need for it.

It is exciting, and a little intimidating to realize how much our notion of the 'solution' to a problem has changed in one century. The story is 'virtual' history because the next two investigators, A. C. Aitken and H. Rutishauser, were not *directly* influenced by Hadamard's dissertation. However they could have been! Aitken rediscovered Hadamard's idea for himself (about 1925) and, driven by the existence of computing machines, though hand driven and mechanical, saw the weakness in the formal solution.

What prompted our choice of title is the intriguing fact that a certain well known quadratic identity among Hankel determinants, (7) in Section 4, that Aitken used to good effect was stated in Hadamard's dissertation, see [8, p. 20, formula (14)], and used in the analysis but not exploited to compute the determinants with minimal effort.

At this point we acknowledge our debt to Peter Henrici who began our story in [9]. Unfortunately he died before the recent investigations that have brought Rutishauser's quotient-difference (qd) algorithm back into the limelight after its eclipse in the 1960s by the QR algorithm.

After describing the LR algorithm and mentioning QR we present new forms of LR and qd that will bring them back onto center stage for both sequential and parallel computation.

These sections serve to round out the study we began in [13].

2 The Task

Consider a square invertible complex matrix B; $B \in \mathbb{C}^{N\times N}$. The eigenvalues $\{\lambda_i\}$ are labelled in decreasing order by magnitude

$$0 < L < |\lambda_N| \leq |\lambda_{N-1}| \leq \ldots \leq |\lambda_1|. \tag{1}$$

The basic goal is to compute B's spectrum from the entries of B.

In order to invoke Hadamard's thesis the basic goal is mapped into a slightly more general problem. Given B and any two (column) vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ in $\mathbb{C}^N$ one may define a rational function f by

$$f(z) := \boldsymbol{u}^*(I_N - zB)^{-1}\boldsymbol{v} \tag{2}$$

$$= \sum_{i=0}^{\infty} a_i z^i, \quad |z| < |\lambda_1|^{-1}, \tag{3}$$

where the row vector $\boldsymbol{u}^*$ is the conjugate transpose of $\boldsymbol{u}$, I_N is the identity matrix in $\mathbb{C}^{N\times N}$, and $z \in \mathbb{C}$.

Note that (2) holds for all $z \in \mathbb{C}$, $z \neq {\lambda_j}^{-1}$, $j = 1, \ldots, N$ while (3) holds only in a neighborhood of the origin. In principle one may compute as many of the Taylor coefficients as desired from the formula

$$a_i = \boldsymbol{u}^* B^i \boldsymbol{v}, \quad i = 0, 1, 2, \ldots \tag{4}$$

The poles of f are the $\{{\lambda_j}^{-1}\}$ and, when they are simple we have the simplest partial fraction representation,

$$f(z) = \sum_{j=1}^{N} \frac{\gamma_j}{1 - z\lambda_j}. \tag{5}$$

To avoid distracting technical complications we assume throughout that the eigenvalues are simple.

Now we can formulate Hadamard's thesis problem. Given the Taylor coefficients $(a_0, a_1, a_2, \ldots)$ of a meromorphic function f find all its poles. The generalization from rational f to meromorphic f requires no new ideas and we are content to restrict attention to the form (2). In 1884 König, see [10], in Germany, had shown how to find λ_1, when $|\lambda_1| > |\lambda_2|$, namely

$$\frac{a_n}{a_{n+1}} \longrightarrow \lambda_1^{-1}, \quad n \longrightarrow \infty.$$

Of course $|\lambda_1^{-1}|$ is the radius of convergence of (3) but, by analytic continuation, the sequence (a_i) determines f everywhere except at its poles.

Six years after König's result was published in 1884 there was still no formula for extracting the larger poles from (a_i) and this was the task that Hadamard selected.

Even a century later students in a complex variable course complain if asked how to compute λ_2^{-1}. The clue is to define $g(z) = (1 - \lambda_1 z)f(z)$, find the Taylor coefficients of g and apply König's result.

3 Hadamard's Solution

By extending Bernoulli's method for approximating zeros of a polynomial using recurrences Hadamard found the key that unlocked the door that leads to the larger poles of f, namely certain Hankel determinants built out of the coefficients (a_i). Define, for nonnegative integers k and n,

$$H_k^n := \det \begin{bmatrix} a_n & a_{n+1} & \cdots & a_{n+k-1} \\ a_{n+1} & a_{n+2} & \cdots & a_{n+k} \\ \vdots & & \vdots & \\ a_{n+k-1} & \cdots & \cdots & a_{n+2k-2} \end{bmatrix}, \qquad (6)$$

$$H_0^n := 1, \quad H_1^n := a_n.$$

Please note the different roles of k and n.

He then expanded (5) in powers of z in order to obtain expressions for each a_i in terms of the $\{\lambda_j\}$ and then substituted the new expressions in an expansion of (6). This yields

Theorem 1 *As $n \longrightarrow \infty$. with f of the form (5),*

$$H_k^n = \text{constant} \cdot (\lambda_1 \cdots \lambda_k)^n \left[1 + O\left(\frac{L}{\lambda_k}\right)^n\right].$$

The original proof is in [8] but an easier one is given in [9].

Corollary 1 *For large enough n, $H_k^n \neq 0$.*

Corollary 2 *For suitable k (i. e. $|\lambda_k| > |\lambda_{k+1}|$),*

$$\frac{H_k^{n+1}}{H_k^n} \longrightarrow \lambda_1 \lambda_2 \cdots \lambda_k, \quad n \longrightarrow \infty.$$

Corollary 3 *If $|\lambda_{k-1}| > |\lambda_k| > |\lambda_{k+1}|$ then*

$$q_k^n := \frac{H_k^{n+1}}{H_k^n} \cdot \frac{H_{k-1}^n}{H_{k-1}^{n+1}} \longrightarrow \lambda_k, \quad n \longrightarrow \infty.$$

Corollary 4 *If f has exactly N poles (counting multiplicities), then*

$$H_{N+1}^n = 0, \quad \text{all } n.$$

We mention, in passing, that when $|\lambda_{k-1}| = |\lambda_k| = |\lambda_{k+1}|$ the situation is not hopeless, merely more complicated. Certain combinations of neighboring q_k^n converge to the coefficients of a polynomial whose zeros are the eigenvalues with modulus $|\lambda_k|$. If any λ_k is multiple Theorem 1 needs to be modified, see [7].

Hadamard left the problem at this point and the remarks we make below are in no sense, to be construed as criticism. In the mathematics of his day there was no question of implementing the process implied by Corollary 3. He had shown exactly how the poles are determined by the coefficients of the Taylor series, a problem of some subtlety whose solution had eluded earlier researchers such as König.

The fact remains that Hadamard's formulae are *useless* for computation in finite precision arithmetic. It is not simply the labor required to compute many determinants. In general the poles $(\lambda_j)^{-1}$, $j > 1$, are very sensitive to small changes in the coefficients (a_n). In other words one must know the (a_n) correct to many decimals in order to determine some of the larger poles to just one or two decimals. Incidentally there is no need to evaluate a determinant from its definition as a signed sum of products of entries. By computing a triangular factorization any $k \times k$ determinant may be found in approximately $k^3/3$ scalar multiplications.

The rest of the story shows how both the (H_k^n) and the (a_n) may be replaced by other quantities that determine the spectrum of B in a less sensitive manner.

4 Aitken's Scheme

During the 1920s Aitken worked on the problem of finding all the zeros of a polynomial by generalizing Bernoulli's method , see [1], and was led to study the Hankel determinants. He discovered a relation which allows the recursive calculation of the H_k^n from the a_i. This is most easily shown by arranging the H_k^n in the following table:

$$
\begin{array}{ccccccc}
1 & & & & & & \\
1 & H_1^0 & & & & & \\
1 & H_1^1 & H_2^0 & & & & \\
1 & H_1^2 & H_2^1 & H_3^0 & & & \\
1 & H_1^3 & H_2^2 & H_3^1 & H_4^0 & & \\
1 & H_1^4 & H_2^3 & H_3^2 & H_4^1 & H_5^0 & \\
\cdot & \cdot & \cdot & \cdot & \cdot & \cdot & \cdot
\end{array}
\qquad (H-\text{Table})
$$

Note that $H_1^n = a_n$ and so the first two columns are known.

In the case under consideration $f(z)$ has only N poles and so the H-table has only $N+1$ columns.

The ratios of successive elements in the kth column of the H-table converge to the product of the largest k eigenvalues by Corollary 2, provided that $|\lambda_k| > |\lambda_{k+1}|$. By Corollary 1 the H-table ultimately exists, i. e. $H_k^n \neq 0$, $k \leq N$, n large enough. However it is possible for an early term to vanish and this would prevent the recursive computation of the table from previous terms. This possibility will be ignored here.

The relation which Aitken found for himself [2], but which had been known to Hadamard, is

$$(H_k^n)^2 - H_k^{n-1} H_k^{n+1} + H_{k+1}^{n-1} H_{k-1}^{n-1} = 0. \tag{7}$$

This is best remembered by taking any H_k^n as $\mathcal{O}$ and labelling its nearest neighbors in the H-table topographically as N, S, W, and E. With such a stencil (7) can be written as

$$\mathcal{O}^2 = N\,S - W\,E. \tag{8}$$

This formula can be used in two different ways:
(a) Start with the first two columns and work from left to right in the H-table, adding one column at a time by repeated use of $E = (N\,S - \mathcal{O}^2)/W$.

(b) If the first two diagonals are known one can compute the lower diagonals successively working along the new diagonal from left to right, using $S = (\mathcal{O}^2 - W\,E)/N$.

Now (b) has the difficulty of finding the first two diagonals and, although he showed how to do this for a polynomial whose coefficients are given, Aitken in [2] favored method (a).

For the computation of eigenvalues the generation of the H-table by (a) or (b), though feasible, still leaves a lot to be desired. With (a) it is necessary to compute $a_n = H_1^n = \boldsymbol{y}^* A^n \boldsymbol{x}$ for as many n as necessary to obtain convergence of H_1^{n+1}/H_1^n to the desired degree of accuracy. With (b) there is the necessity of computing a_i for $i = 0, 1, \ldots, 2N-2$ and then the determinants H_k^0 and H_k^1 for $k = 1, 2, \ldots, N$. If Gaussian elimination were used then, whilst calculating H_N^0 and H_N^1, all other H_i^0 and H_i^1 $(i = 1, 2, \ldots N-1)$, being leading principal minors, could be found. This technique, however, would not permit the use of interchanges in the Gaussian elimination and so extra precision would be required to ensure adequate accuracy. (In all, $2N^3$ multiplications are needed for the a_i and $\frac{2}{3}N^3$ double precision multiplications for the H_i^0 and H_i^1.) Thus both (a) and (b) demand a formidable amount of computation before the remainder of the H-table can be found.

Since Hadamard knew the quadratic relation (7) he could easily have anticipated all of Aitken's results but, as we have stressed, the motivation was lacking.

It was Rutishauser's achievement to produce a related table which is numerically preferable.

5 The qd Algorithm

What changes could be made in the H-scheme which might reduce the amount of initial computation? There is one natural observation: the H_k^n themselves are of less interest than the

$$q_k^n = \frac{H_k^{n+1} H_{k-1}^n}{H_k^n H_{k-1}^{n+1}},$$

since for $|\lambda_1| > |\lambda_2| > \ldots |\lambda_N|$, as $n \longrightarrow \infty$

$$q_k^n \longrightarrow \lambda_k.$$

As Henrici says in [9]: 'It is remarkable that in the computation of the q_k^n, the determinants H_k^n do not have to be used if a set of auxiliary quantities is introduced.' Their introduction is presented as a 'fait accompli' by Rutishauser in [14]. Certainly this set of quantities arises naturally from the continued fraction expansion of $f(z)$ (see [9, p. 36]) but this author feels that a motivation for the choice of the auxiliary quantities, called e_k^n, can be made in the present context. This is our 'virtual' history. We show how Rutishauser might have derived qd from Aitken's H-table.

The key is to recast the basic recursion relation (8) so that it shows explicitly the connection between q_k^n and q_k^{n+1}. To do this clearly, consider a typical portion of the H-table obtained by placing the following stencil on the table so that D is on H_k^n.

$$\begin{array}{cccccc} \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot & A & \cdot & \cdot & \cdot \\ \cdot & B & C & D & \cdot & \cdot \\ \cdot & \cdot & E & F & G & \cdot \\ \cdot & \cdot & \cdot & H & I & \cdot \\ \cdot & \cdot & \cdot & \cdot & \cdot & \cdot \end{array}$$

Then

$$q_k^n = \frac{AF}{CD}, \quad q_k^{n+1} = \frac{CH}{EF}.$$

To include all these elements two applications of (8) are needed, centered at C and F respectively, namely

$$C^2 = AE - BD, \quad F^2 = DH - EG.$$

Now turn first terms on the right hand sides into q_k^n and q_k^{n+1} respectively, obtaining

$$\frac{CF}{DE} = \frac{AF}{CD} - \frac{BF}{CE} \quad \frac{CF}{DE} = \frac{CH}{EF} - \frac{CG}{DF}. \tag{9}$$

Reference to the stencil shows that BF/CE and CG/DF are of the same form and naturally ask to be named. If CG/DF $(= H_{k-1}^{n+1} H_{k+1}^n / H_k^n H_k^{n+1})$ is called e_k^n then BF/CE will be e_{k-1}^{n+1}. Transposing the second terms on the right hand sides of (9) gives the following simple relation between the q and the e,

$$q_k^n + e_k^n = q_k^{n+1} + e_{k-1}^{n+1}. \tag{10}$$

The quadratic relation (8) has become a linear relation (10).

Another relation between the q and the e is required to determine all the q_k^n and e_k^n from the q_1^n $(= a_{n+1}/a_n)$. It follows directly from the definitions of the q and e. A simple derivation comes from observing that

$$\frac{CH}{EF} \cdot \frac{EI}{FH} = \frac{CI}{F^2} = \frac{DI}{GH} \cdot \frac{CG}{DF}.$$

In other words

$$q_k^{n+1} \cdot e_k^{n+1} = q_{k+1}^n \cdot e_k^n. \tag{11}$$

The qd Scheme

Formula (10) shows that e_k^n is the *difference* of the *quotients* q_k^{n+1} and q_k^n *modified* by e_{k-1}^{n+1}. This indicates a little why Rutishauser named the following triangular table of q's and e's the *Quotient-Difference Scheme.* Typical cases of (10) and (11) are shown pictorially in Figure 1 and explain why E. Stiefel named them the rhombus rules. (10) yields rhombi centered on q-columns, (11) yields rhombi centered on e-columns.

By alternate application of the rules the table may be generated, in theory, form either

(a) the first q-column moving right, or

(b) the first diagonal moving down.

Now, however, method (a) is quite impractical since rule (10) shows that

$$e_1^n = q_1^{n+1} - q_1^n.$$

For $|\lambda_1| > |\lambda_2|$, $q_1^n \longrightarrow \lambda_1$, as $n \longrightarrow \infty$. As n increases e_1^n becomes the difference of two almost equal numbers and, for computation with a fixed number of digits for each number, it will have fewer and fewer significant figures. Thus (e_1^{n+1}/e_1^n) and hence q_2^n will have little or no accuracy. Briefly, one says that method (a) is 'unstable'. Method (b), on the other hand, uses (10) in the form

$$q_k^{n+1} = q_k^n + e_{k+1}^n - e_{k-1}^{n+1}$$

which is ultimately completely safe when e_{k+1}^n and $e_{k-1}^{n+1} \longrightarrow 0$.

The danger in method (b) is that in the early stages, for small n, large values of e_{k-1}^{n+1} may occur which will cause loss of significant digits in the q_k^{n+1}.

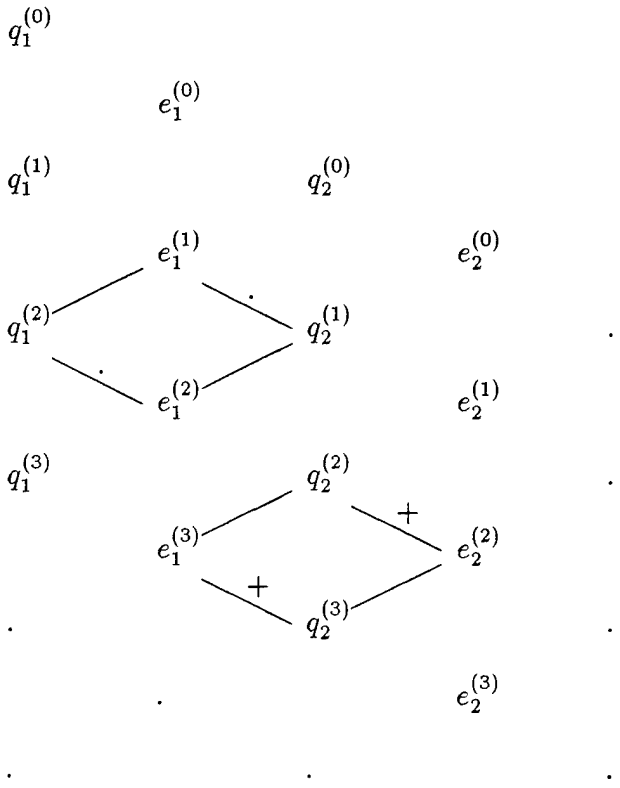


Figure 1: The qd table

However with the qd scheme there is no alternative but to use (b), the so called progressive method. Rutishauser in [16] has analyzed the effect of one small q_k^{n+1} and the resulting large value of e_k^{n+1} (in using (11)). He observed that only negative powers of e_k^{n+1} occur in diagonal $n+3$ and below. Thus the disturbance affects only the next two diagonals and he proposed some auxiliary modifications to be made in calculating some of the elements in diagonals $n+1$ and $n+2$. For diagonal $n+3$ and below the straightforward application of the rhombus rules is again sufficient. The recently discovered *differential* form of qd, see Section 8, goes a long way towards avoiding the need for these modifications.

6 Calculation of the First Diagonal

At this point it is worth observing that

$$q_1^n = \frac{H_1^{n+1}}{H_1^n} = \frac{\boldsymbol{u}^* B^{n+1} \boldsymbol{v}}{\boldsymbol{u}^* B^n \boldsymbol{v}}$$

and so the descent of column 1 of the qd scheme corresponds to the well known power method [17].

How can the first diagonal of the qd scheme be found indirectly? The answer is, surprisingly, by use of Lanczos method of 'minimized iterations', see [11], with initial vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ (from the definition of $f(z)$ in Section 2). Thus, as Henrici points out, the qd scheme links the power method to the method of Lanczos.

The reader is referred to [11] and [9] for proofs and details of the algorithm. It produces a tridiagonal matrix J similar to B and of the form (for $N = 5$)

$$J = \begin{bmatrix} \alpha_1 & 1 & & & \\ \beta_1 & \alpha_2 & 1 & & \\ & \beta_2 & \alpha_3 & 1 & \\ & & \beta_3 & \alpha_4 & 1 \\ & & & \beta_4 & \alpha_5 \end{bmatrix}$$

By the LDU theorem [3, p. 20], J (or $J + sI$) may be written as the product LR

$$L = \begin{bmatrix} 1 & & & & \\ e_1 & 1 & & & \\ & e_2 & 1 & & \\ & & e_3 & 1 & \\ & & & e_4 & 1 \end{bmatrix}, \quad R = \begin{bmatrix} q_1 & 1 & & & \\ & q_2 & 1 & & \\ & & q_3 & 1 & \\ & & & q_4 & 1 \\ & & & & q_5 \end{bmatrix}.$$

Is is easily verified that for $k = 1, 2, \ldots N$,

$$\begin{aligned} \alpha_k &= q_k + e_{k-1}, \quad e_0 = 0, \\ \beta_k &= q_k e_k, \qquad\quad e_N = 0. \end{aligned}$$

Henrici shows [9, p. 33] that these q_k and e_k are the elements of the first diagonal of the qd scheme. In fact he shows that the nth diagonal can be

obtained in the same way by using Lanczos method with initial vectors $B^n\boldsymbol{u}$ and $\boldsymbol{v}$.

$$\alpha_1^n = \frac{\boldsymbol{u}^* B^{n+1}\boldsymbol{v}}{\boldsymbol{u}^* B^n \boldsymbol{v}} = q_1^n$$

as it should.

When $|\lambda_1| > |\lambda_2| > \ldots |\lambda_N|$ then $q_k^n \longrightarrow \lambda_k$, $e_k^n \longrightarrow 0$, as $n \longrightarrow \infty$. When there are eigenvalues with the same modulus then the corresponding q and e columns may not converge. Fortunately this does not impair the usefulness of the algorithm. Discussion of what to do in this case is most simply presented in connection with the LR transformation and will appear in the next section.

7 The LR Transformation

In Section 6 it was shown that the elements q_i^0 and e_i^0 of the first diagonal of the qd scheme are obtained from the LR decomposition of a certain tridiagonal matrix J, now to be called J_0, whose nonzero elements in column i are 1, α_i, β_i. this implies that, for $k = 1, 2, \ldots, N$,

$$\begin{aligned} q_k^0 + e_{k-1}^0 &= \alpha_k^0, \qquad e_0^0 = 0, \\ q_k^0 e_k^0 &= \beta_k^0, \qquad \beta_N^0 = 0. \end{aligned} \tag{12}$$

The rhombus rules, (10) and (10), which yield the next diagonal have a very interesting interpretation. Recall from the previous section that

$$L_0 = \begin{bmatrix} 1 & & & & \\ e_1^0 & 1 & & & \\ & e_2^0 & 1 & & \\ & & \cdot & \cdot & \\ & & & e_4^0 & 1 \end{bmatrix}, \qquad R_0 = \begin{bmatrix} q_1^0 & 1 & & & \\ & q_2^0 & 1 & & \\ & & q_3^0 & 1 & \\ & & & \cdot & \cdot \\ & & & & q_5^0 \end{bmatrix}.$$

From (10) and (11) and the rules of matrix multiplication follow the result

$$\begin{aligned} q_k^1 + e_{k-1}^1 &= q_k^0 + e_k^0 = (R_0 L_0)_{kk} = \alpha_k^1, \quad \text{(say)}, \\ q_k^1 e_k^1 &= q_{k+1}^0 e_k^0 = (R_0 L_0)_{k,k+1} = \beta_k^1, \quad \text{(say)}. \end{aligned} \tag{13}$$

It is easily verified that R_0L_0 is a tridiagonal matrix of the same form as J_0 and may be called J_1. Now comparison of (13) with (12) shows that the second diagonal of the qd scheme (q_i^1, e_i^1) contains precisely the elements of the LR decomposition of R_0L_0 $(= J_1)$. Thus

$$J_0 \text{ is factorized into } L_0R_0,$$

The factors are multiplied in reverse order to give $R_0L_0 = J_1$,

$$J_1 \text{ is factorized into } L_1R_1.$$

Similarly the third diagonal of the qd scheme contains the elements of the LR decomposition of R_1L_1 $(= J_2)$, and so on for all the diagonals. In fact, for each positive integer n, equations (12), with superscript n, define the elements α_i^n, β_i^n of a tridiagonal matrix J_n. The J_n may be read, as it were, 'between the (diagonal) lines' of the qd scheme. Clearly $J_{n+1} = L_n^{-1}J_nL_n$ and if the eigenvalues λ_i of J_0 (or the original A) have distinct moduli then, $n \longrightarrow \infty$,

$$J_n \longrightarrow R_\infty \quad \text{(a triangular matrix with } q_k^\infty = \lambda_k\text{)}.$$

Thus the qd scheme implicitly defines a convergent sequence of tridiagonal matrices similar to J_0. this was a brilliant insight by Rutishauser.

For tridiagonal matrices which are symmetric positive definite the qd algorithm may be used since the danger (instability) mentioned in Section 4 cannot occur. This technique is from three to six times as fast as the much used Sturm sequence method, see [6], See Rutishauser [16] for full details.

Dense Matrices

This interpretation of the qd scheme suggests a similar algorithm for full nonsingular matrix B_1. By the LDU theorem, [3, p. 20], B_1 (or $B_1 + sI$) may be factorized into the product of left triangular matrix L_1 with 1's on the diagonal and right triangular matrix R_1. The LR transformation is defined by Rutishauser in [15] as follows. For $k = 1, 2, \ldots$

$$\begin{cases} \text{Factorize } B_k \text{ into } L_kR_k, \\ \text{Form } R_kL_k \text{ and call it } B_{k+1}. \end{cases}$$

Clearly $B_{k+1} = L_k^{-1}B_kL_k$ and it may be asked when this sequence $\{B_k\}$ converges to a triangular matrix.

The answer, see [15], is contained in the following

Theorem 2 *If* $B_1 = U\, diag(\lambda_1, \cdots, \lambda_N)U^{-1}$ *and*
(a) $|\lambda_1| > |\lambda_2| > \cdots > |\lambda_N|$,
(b) The leading principal minors of U *and* U^{-1} *are nonzero, then* $\lim B_k = B_\infty$ *exists and is upper triangular with* $(B_\infty)_{ii} = \lambda_i$.

If the $|\lambda_i|$ are not distinct then, loosely speaking, B_k may be said to converge to block triangular form. More precisely suppose that $|\lambda_1| = |\lambda_2| = \cdots = |\lambda_p| > |\lambda_{p+1}|$. Strictly speaking B_∞ may not exist but, $k \longrightarrow \infty$, B_k becomes reduced. The elements in the first p columns may not converge but the characteristic polynomial of the leading principal submatrix of order p does converge to the monic polynomial with roots $\lambda_1, \ldots, \lambda_p$. The complementary submatrix has eigenvalues which converge to $\lambda_{p+1}, \ldots, \lambda_N$. If any of these eigenvalues have equal modulus then this submatrix will become reduced also. Thus as $k \longrightarrow \infty$ the submatrix blocks which become isolated along the diagonal correspond to groups of eigenvalues of equal modulus. In the qd scheme, analogously, when successive q and e columns fail to converge, it turns out that certain polynomials, formed from the qs and e's in those columns and in the nth diagonal, do converge as $n \longrightarrow \infty$.

In the important case of real matrices with complex conjugate pairs of eigenvalues, B_k may be expected to have along the diagonal, for large k, isolated 2×2 real submatrices which yield the eigenvalues very conveniently. This was the reason for stating in the previous section that the occurrence of eigenvalues of equal modulus does not impair the qd algorithm.

Shifts of origin may be used with both the qd algorithm and the LR algorithm to hasten convergence. Thus for scalars σ_j one defines

$$\begin{aligned} B_j - \sigma_j I &= L_j R_j, \\ B_{j+1} &:= R_j L_j + \sigma_j I. \end{aligned}$$

A good strategy for choosing $\{\sigma_j\}$ is important in practice but that is not the focus of this paper. See [15] and [4] for more on shifts.

One might think of the LR algorithm as the final transformation of Hadamard's scheme into a practical algorithm. However the search is not over yet.

The possible element growth that can occur in LR prompted the search for more stable schemes. About 1960 J. G. F. Francis presented his QR algorithm applied to matrices in Hessenberg form and this has been the

method of choice for computing eigenvalues for over 30 years. It does not preserve bandwidth for nonsymmetric matrices.

8 Differential qd

The two rhombus rules that govern the *qd*-array are invoked in alternating fashion to compute the next line of the array. Shifts σ may be introduced and the result is Rutishauser's progressive *qd* algorithm. This may be written symbolically as

$$qds : (q, e) \xrightarrow{\sigma} (\hat{q}, \hat{e}).$$

The code is

$$\begin{aligned}
&\hat{q}_1 = q_1 + e_1 - \sigma \\
&\text{for } j = 1, 2, \ldots, N-1 \\
&\quad \hat{e}_j = e_j * (q_{j+1}/\hat{q}_j) \\
&\quad \hat{q}_{j+1} = q_{j+1} + e_{j+1} - \hat{e}_j - \sigma
\end{aligned}$$

Here $e_N = 0$ and $\hat{e}_N = 0$.

In terms of the LR algorithm of the previous section, *qds* yields the factorization

$$UL - \sigma I = \hat{L}\hat{U}. \tag{14}$$

Now we look at this triangular factorization again, in closer detail, to derive a differential algorithm that computes the same quantities with slightly more arithmetic effort than *qds*.

Look at the reduction of $UL - \sigma I$ to $\hat{U}$ at an intermediate step. Thus the array is

$$\begin{bmatrix}
\cdot & \cdot & \cdot & & & \\
0, & \hat{q}_{k-1}, & 1 & & & \\
 & 0, & \hat{q}_k, & 1 & & \\
 & & e_k q_{k+1}, & q_{k+1} + e_{k+1} - \sigma, & 1 & \\
 & & & e_{k+1} q_{k+2}, & *, & 1
\end{bmatrix}.$$

The key is to write the new pivot $\hat{q}_k$ as a sum of $d_k + e_k$, introducing a new array d. The next step in elimination is

$$\hat{q}_k \;=\; d_k + e_k,$$

$$\begin{aligned} \hat{e}_k &= e_k q_{k+1}/\hat{q}_k, \\ d_{k+1} + e_{k+1} &= q_{k+1} + e_{k+1} - \sigma - q_{k+1} e_k/(d_k + e_k). \end{aligned}$$

The beautiful feature is that e_{k+1} is removed from each side analytically without even being added to q_{k+1}. Simplifying each side yields

$$d_{k+1} = d_k(q_{k+1}/\hat{q}_k) - \sigma.$$

This gives the *dqds* algorithm (differential qd with shifts) introduced by Fernando and Parlett [4].

dqds
$$\begin{aligned} &\hat{d}_1 = q_1 - \sigma \\ &\text{for } k = 1, 2, \ldots, N-1 \\ &\quad \hat{q}_k = d_k + e_k \\ &\quad \hat{e}_k = e_k(q_{k+1}/\hat{q}_k) \\ &\quad d_{k+1} = d_k(q_{k+1}/\hat{q}_k) - \sigma \\ &\hat{q}_N = d_N. \end{aligned}$$

Because of the repeated expression $(q_{k+1}/\hat{q}_k)$ the algorithm may be implemented with only one division in the inner loop.

Should $\hat{q}_k = 0$ the algorithm will break down. Nevertheless in the absence of such a breakdown, the algorithm enjoys a remarkable mixed stability property. When executed in finite precision it is only necessary to change appropriately the input (q, e) and the output $(\hat{q}, \hat{e})$ by 1, 2, or 3 units in the last place of each variable in order to have an exact *dqds* transformation. See [4] for the full story.

One attractive consequence is that when $q, e, \hat{q}$ and $\hat{e}$ are positive then the eigenvalues of UL and $\hat{U}\hat{L}$ agree to high accuracy, however small some of those eigenvalues may be.

The additional variables $\{d_j\}$ seemed like a blemish at first but it turns out that they yield useful approximations to the smallest eigenvalue and so improve shift selection.

9 The Implicit LR Algorithm

The LR algorithm introduced by Rutishauser produces a sequence of matrices $(B_0, B_1, B_2, \ldots)$ on the assumption that triangular factorization is permitted

at each step. Thus $B_j = L_j U_j$. For several reasons it would be preferable to discard the B_j in favor of the pairs L_j, U_j. In order to do that it is necessary to derive L_{j+1}, U_{j+1} from L_j, U_j without forming $B_{j+1} := U_j L_j$ explicitly. To avoid subscripts we write

$$\hat{L}\hat{U} = \hat{B} := UL. \tag{15}$$

Appeal to the two-sided Gram-Schmidt process shows that (15) implies the existence of unique matrices F, G such that

$$L = G\hat{U}, \quad U = \hat{L}F, \quad FG = I. \tag{16}$$

The practical point is that when L and U are bidiagonal then F and G may be generated implicitly as the product of very simple matrices. Even when L and U are dense the F and G matrices may be generated as products of elementary matrices of the form $(I - \boldsymbol{x}\boldsymbol{y}^t)$.

Shifts may be introduced to accelerate convergence and the new equations are

$$L = G\hat{U}, \quad U - \sigma L^{-1} = \hat{L}F, \quad FG = I. \tag{17}$$

It turns out that the presence of L^{-1} does not greatly complicate the representation of F and G. For more information see [5].

This algorithm is attractive when B has narrow bandwidth and one can avoid reducing to tridiagonal form.

10 Parallel Features

The simplicity of the *dqds* algorithm facilitates its adaptation to modern computer architectures, particularly to systems with many arithmetic units. The algorithm sketched below is presented for its intellectual interest. It is very fast but is also unstable, [12]. Nevertheless the idea is worth knowing.

Consider the *dqds* algorithm again and observe that if the d_j, $j = 1, \ldots, N$ were all known then the new $\hat{q}$ and $\hat{e}$ values may be obtained in fully parallel form with each processor responsible for a given subset of array elements. Let $\boldsymbol{v}\uparrow$ denote the array obtained by pushing up, by one index, the entries of $\boldsymbol{v}$ and inserting 0 for the last position. As usual $e(N) = \hat{e}(N) = 0$.

$$\begin{aligned} \text{Parallel dqds:} \quad & \hat{\boldsymbol{q}} = \boldsymbol{d} + \boldsymbol{e}, \\ & \hat{\boldsymbol{e}} = \boldsymbol{e} * ((\boldsymbol{q}\uparrow) \div \hat{\boldsymbol{q}}). \end{aligned}$$

The hard task is to compute $\boldsymbol{d}$ fast. This may be done in $2\log_2 N$ time despite the fact that he recurrence seems intrinsically nonlinear and sequential;

$$d_{k+1} = q_{k+1}d_k/(d_k + e_k) - \sigma. \tag{18}$$

The form changes radically when d_k is written as a quotient $d_k = f_k/g_k$. Then (18) becomes

$$\frac{f_{k+1}}{g_{k+1}} = \frac{f_k(q_{k+1} - \tau) - \sigma e_k g_k}{f_k + e_k g_k}$$

or, in vector notation,

$$\begin{aligned} \begin{pmatrix} f_{k+1} \\ g_{k+1} \end{pmatrix} &= \begin{bmatrix} q_{k+1} - \sigma, & -\sigma e_k \\ 1, & e_k \end{bmatrix} \begin{pmatrix} f_k \\ g_k \end{pmatrix} = W_k \begin{pmatrix} f_k \\ g_k \end{pmatrix} \\ &= W_k W_{k-1} \cdots W_1 \begin{pmatrix} q_1 - \sigma \\ 1 \end{pmatrix}, \end{aligned} \tag{19}$$

for $k = 1, \ldots, N-1$. Note that all W_j are known in advance, from $\boldsymbol{q}, \boldsymbol{e}$ and σ, and we want *all* the partial products.

With N, or $N/2$, processors this task can be accomplished in only $2\log_2 N$ time! The technique is known as 'parallel prefix' among computer scientists. When $\sigma = 0$ the W_j are lower triangular and the products are simplified.

The idea of the algorithm is best understood by a diagram, see Figure 2. On the other hand it can be specified in a very short procedure. Let $m = \log_2 N$ and initialize an array Y to $Y_j = W_j$, $j = 1, \ldots, N$. Upon exit Y_k holds $W_k W_{k-1} \cdots W_1$.

In pseudo code we need only two lines.

for $j = 1^0, 2^2, \ldots, 2^{m-1}$, [**for** $i = 2j, N, 2j$ in parallel $Y(i) = Y(i) * Y(i-j)$]
for $j = 2^{m-2}, 2^{m-3}, \ldots, 2^0$, [**for** $i = 3j, N, 2j$ in parallel $Y(i) = Y(i) * Y(i-j)$]

The third variable, $2j$ in the **for** i expression is called the stride; $i = 2j, 4j, 6j, \ldots, N$. The first line of the algorithm is called going down the tree and the second line is called going up the tree.

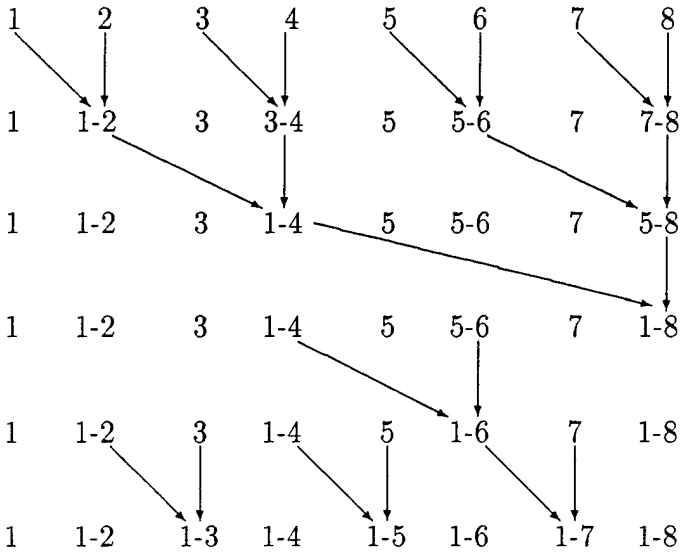


Figure 2: Parallel prefix

11 Conclusion

In the 1920s A. C. Aitken rediscovered much of the material in Hadamard's thesis and in addition, worried about speed versus accuracy when computing the H-table. We showed here (Section 5) that Rutishauser's qd-table can be seen as a reformulation of the H-table to give a more robust array. In fact Rutishauser's motivation came from continued fractions and not from Aitken's work.

It is quite well known that the qd-algorithm enabled Rutishauser to discover the LR algorithm. For tridiagonals LR may be viewed as a reformulation of qd and thus as a descendent of the H-table. Instead of focusing on QR as a stable variation of LR, as in [13], we have chosen to present recent variants on qd and LR which seem preferable to the original ones. The new techniques preserve bandwidth and can afford to have a much more sophisticated shift strategy than QR. It remains to be seen whether they will displace QR. In any case they may be perceived as computationally desirable descendants of Hadamard's H-table.

References

[1] A. C. Aitken. On Bernoulli's numerical solution of algebriac equations. *Proc. Roy. Soc. Edinburgh, Section A*, 46:289–305, 1926.

[2] A. C. Aitken. Further numerical studies in algebraic equations and matrices. *Proc. Roy. Soc. Edinburgh, Section A*, 51:80–90, 1931.

[3] V. N. Faddeeva. *Computational Methods of Linear Algebra.* Dover Publications, London, 1959.

[4] K. V. Fernando and B. N. Parlett. Accurate singular values and differential qd algorithms. *Numer. Math.*, 67:191–229, 1994.

[5] K. V. Fernando and B. N. Parlett. Implicit Cholesky algorithm for singular values and vectors. *Int. J. of Numerical Linear Algebra and Applications*, 2:507–531, 1995.

[6] W. Givens. Numerical computation of the characteristic values of a real symmetric matrix. Technical Report ORNL-1574, Oak Ridge Nat. Lab., 1954.

[7] M. Golomb. Zeros and poles of functions defined by taylor series. *Bull. Am. Math. Soc.,*, 49:581–592, 1943.

[8] M. J. Hadamard. *Essai sur l'étude des fonctions données par leur devéloppement de Taylor.* Thèses pour obtenir le grade de docteur en Sciences Mathematiques, Gauthier-Villars et Fils, 55 Quai des Grands-Augustins, Paris, 1891.

[9] P. Henrici. The quotient-difference algorithm. *Nat. Bur. Standards Appl. Math. Series*, 19:23–46, 1958.

[10] Julius König. Ueber eine Eigenschaft der Potenzreihen. *Math. Ann.*, 23:447–449, 1884.

[11] C. Lanczos. An iteration method for the solution of the eigenvalue problem of linear differential and integral operators. *J. Research NBS.*, 45:255–282, 1950.

[12] R. Mathias. The stability of parallel prefix matrix multiplication - with applications to tridiagonal matrices. Preprint, Dept. of Mathematics, College of William and Mary, Williamsburg, VA 23187, 1994.

[13] Beresford N. Parlett. The development and use of methods of LR type. *SIAM Review*, 6, 1964.

[14] H. Rutishauser. Der Quotienten-Differenzen-Algorithmus. *Z. Angew. Math. Phys.*, 5:233–251, 1954.

[15] H. Rutishauser. Solution of eigenvalue problems with the LR-transformation. *Nat. Bur. Standards Appl. Math. Series*, 49:47–81, 1958.

[16] H. Rutishauser. Stabile Sonderfalle des QD-Algorithmus. *Num. Math.*, 5:95, 1963.

[17] J. H. Wilkinson. The use of iterative methods for finding the latent roots and vectors of matrices. *Math. Tables and Other Aids to Comput.*, 9:184–191, 1955.

Legal Notice